

CLAIMS

1 1. A method for checking a model, which defines states
2 of a system under study and a transition relation among
3 the states, the method comprising:

4 specifying a property that applies to a target set
5 that comprises at least one target state among the states
6 of the system under study;

7 beginning from an initial set of at least one
8 initial state among the states of the system, computing
9 successive reachable sets comprising the states of the
10 system that are reachable from the initial set;

11 finding an intersection between one of the reachable
12 sets and the target set; and

13 computing a plurality of mutually-disjoint traces
14 from the at least one target state in the intersection
15 through the states in the reachable sets to the at least
16 one initial state.

1 2. A method according to claim 1, wherein specifying
2 the property comprises specifying a condition that is
3 expected to be true over all of the reachable states of
4 the system under study, and wherein the condition is
5 false in the at least one target state.

1 3. A method according to claim 1, wherein specifying
2 the property comprises specifying a condition
3 representing a desired behavior of the system under
4 study, such that the condition is fulfilled in the at
5 least one target state.

1 4. A method according to claim 1, wherein computing the
2 successive reachable sets comprises testing the property

3 while computing the sets, and ceasing to compute the sets
4 when the intersection is found.

1 5. A method according to claim 1, wherein computing the
2 successive reachable sets comprises:

3 determining a first one of the reachable sets,
4 disjoint from the initial set, such that all of the
5 states in the first one of the reachable sets are reached
6 from the at least one initial state in a first cycle of
7 the transition relation; and

8 determining the successive reachable sets, following
9 the first one of the reachable sets, such that all of the
10 states in each of the reachable sets are reached from the
11 states in a preceding one of the reachable sets in a
12 successive cycle of the transition relation, and so that
13 each of the successive reachable sets is disjoint from
14 the initial set and from the other reachable sets
15 determined before it.

1 6. A method according to claim 5, wherein computing the
2 traces comprises, for each trace among the multiple
3 traces, selecting one of the states from each of the
4 successive reachable sets.

1 7. A method according to claim 6, wherein selecting the
2 one of the states comprises, for each of the selected
3 states, choosing a predecessor state among the states in
4 a preceding one of the reachable sets until the state on
5 the trace in the first one of the reachable sets is
6 found, and choosing the predecessor state in the initial
7 set to the state in the first one of the reachable sets.

1 8. A method according to claim 7, wherein choosing the
2 predecessor state comprises, on each of the traces

41484S2

3 computed after a first one of the traces, choosing the
4 predecessor state so as to maximize a distance of the
5 trace from the other traces already computed.

1 9. A method according to claim 1, wherein computing the
2 traces comprises selecting the states on each trace among
3 the multiple traces so as to maximize a distance of the
4 trace from the other traces already computed.

1 10. A method according to claim 9, wherein each of the
2 states is represented by a binary decision diagrams
3 (BDD), and wherein selecting the states on each trace
4 comprises maximizing the distance between the BDD
5 representing the state to be selected and the BDD
6 representing the states on the other traces.

1 11. A method according to claim 10, wherein maximizing
2 the distance comprises:

3 taking a left trial state and a right trial state on
4 left and right branches, respectively, of the BDD
5 representing the state to be selected; and

6 choosing the trial state that has a larger Hamming
7 distance from the BDD representing the states on the
8 other traces.

1 12. Model checking apparatus, comprising a model
2 processor, which is arranged to receive a model defining
3 states of a system under study and a transition relation
4 among the states, and to receive a specification of a
5 property that applies to a target set comprising at least
6 one target state among the states of the system under
7 study, the processor being further arranged to compute,
8 beginning from an initial set of at least one initial
9 state among the states of the system, successive

10 reachable sets comprising the states of the system that
11 are reachable from the initial set, to find an
12 intersection between one of the reachable sets and the
13 target set, and to compute a plurality of
14 mutually-disjoint traces from the at least one target
15 state in the intersection through the states in the
16 reachable sets to the at least one initial state.

1 13. Apparatus according to claim 12, wherein the
2 property comprises a condition that is expected to be
3 true over all of the reachable states of the system under
4 study, and wherein the condition is false in the at least
5 one target state.

1 14. Apparatus according to claim 12, wherein the
2 property comprises a condition representing a desired
3 behavior of the system under study, such that the
4 condition is fulfilled in the at least one target state.

1 15. Apparatus according to claim 12, wherein the
2 processor is arranged to test the property while
3 computing the sets, and ceases to compute the sets when
4 the intersection is found.

1 16. Apparatus according to claim 12, wherein the
2 processor is arranged to compute a first one of the
3 reachable sets, disjoint from the initial set, such that
4 all of the states in the first one of the reachable sets
5 are reached from the at least one initial state in a
6 first cycle of the transition relation, and further to
7 compute the successive reachable sets, following the
8 first one of the reachable sets, such that all of the
9 states in each of the reachable sets are reached from the
10 states in a preceding one of the reachable sets in a

41484S2

11 successive cycle of the transition relation, and so that
12 each of the successive reachable sets is disjoint from
13 the initial set and from the other reachable sets
14 determined before it.

1 17. Apparatus according to claim 16, wherein the
2 processor is arranged to compute the traces by selecting,
3 for each trace among the multiple traces, one of the
4 states from each of the successive reachable sets.

1 18. Apparatus according to claim 17, wherein the
2 processor is arranged to compute the traces by choosing,
3 for each of the selected states, a predecessor state
4 among the states in a preceding one of the reachable sets
5 until the state on the trace in the first one of the
6 reachable sets is found, and choosing the predecessor
7 state in the initial set to the state in the first one of
8 the reachable sets.

1 19. Apparatus according to claim 18, wherein the
2 processor is arranged to choose the predecessor state on
3 each of the traces computed after a first one of the
4 traces so as to maximize a distance of the trace from the
5 other traces already computed.

1 20. Apparatus according to claim 12, wherein the
2 processor is arranged to compute the traces by selecting
3 the states on each trace among the multiple traces so as
4 to maximize a distance of the trace from the other traces
5 already computed.

1 21. Apparatus according to claim 20, wherein each of the
2 states is represented by a binary decision diagrams
3 (BDD), and wherein the processor is arranged to select
4 the states on each trace so as to maximize the distance

5 between the BDD representing the state to be selected and
6 the BDD representing the states on the other traces.

1 22. Apparatus according to claim 21, wherein the
2 processor is arranged to maximize the distance by taking
3 a left trial state and a right trial state on left and
4 right branches, respectively, of the BDD representing the
5 state to be selected, and choosing the trial state that
6 has a larger Hamming distance from the BDD representing
7 the states on the other traces.

1 23. A computer software product, comprising a
2 computer-readable medium in which program instructions
3 are stored, which instructions, when read by a computer,
4 cause the computer to receive a model defining states of
5 a system under study and a transition relation among the
6 states, and to receive a specification of a property that
7 applies to a target set comprising at least one target
8 state among the states of the system under study, the
9 instructions further causing the computer to compute,
10 beginning from an initial set of at least one initial
11 state among the states of the system, successive
12 reachable sets comprising the states of the system that
13 are reachable from the initial set, to find an
14 intersection between one of the reachable sets and the
15 target set, and to compute a plurality of
16 mutually-disjoint traces from the at least one target
17 state in the intersection through the states in the
18 reachable sets to the at least one initial state.

1 24. A product according to claim 23, wherein the
2 property comprises a condition that is expected to be
3 true over all of the reachable states of the system under

4 study, and wherein the condition is false in the at least
5 one target state.

1 25. A product according to claim 23, wherein the
2 property comprises a condition representing a desired
3 behavior of the system under study, such that the
4 condition is fulfilled in the at least one target state.

1 26. A product according to claim 23, wherein the
2 instructions cause the computer to test the property
3 while computing the sets, and to cease to compute the
4 sets when the intersection is found.

1 27. A product according to claim 23, wherein the
2 instructions cause the computer to compute a first one of
3 the reachable sets, disjoint from the initial set, such
4 that all of the states in the first one of the reachable
5 sets are reached from the at least one initial state in a
6 first cycle of the transition relation, and further to
7 compute the successive reachable sets, following the
8 first one of the reachable sets, such that all of the
9 states in each of the reachable sets are reached from the
10 states in a preceding one of the reachable sets in a
11 successive cycle of the transition relation, and so that
12 each of the successive reachable sets is disjoint from
13 the initial set and from the other reachable sets
14 determined before it.

1 28. A product according to claim 27, wherein the
2 instructions cause the computer to compute the traces by
3 selecting, for each trace among the multiple traces, one
4 of the states from each of the successive reachable sets.

1 29. A product according to claim 28, wherein the
2 instructions cause the computer to compute the traces by

41484S2

3 choosing, for each of the selected states, a predecessor
4 state among the states in a preceding one of the
5 reachable sets until the state on the trace in the first
6 one of the reachable sets is found, and choosing the
7 predecessor state in the initial set to the state in the
8 first one of the reachable sets.

1 30. A product according to claim 29, wherein the
2 instructions cause the computer to choose the predecessor
3 state on each of the traces computed after a first one of
4 the traces so as to maximize a distance of the trace from
5 the other traces already computed.

1 31. A product according to claim 23, wherein the
2 instructions cause the computer to compute the traces by
3 selecting the states on each trace among the multiple
4 traces so as to maximize a distance of the trace from the
5 other traces already computed.

1 32. A product according to claim 31, wherein each of the
2 states is represented by a binary decision diagrams
3 (BDD), and wherein the instructions cause the computer to
4 select the states on each trace so as to maximize the
5 distance between the BDD representing the state to be
6 selected and the BDD representing the states on the other
7 traces.

1 33. A product according to claim 21, wherein the
2 instructions cause the computer to maximize the distance
3 by taking a left trial state and a right trial state on
4 left and right branches, respectively, of the BDD
5 representing the state to be selected, and choosing the
6 trial state that has a larger Hamming distance from the
7 BDD representing the states on the other traces.